INTERWOVEN

DataDeploy™ User's Guide

Release 5.5.1

INTERWOVEN

Interwoven, Inc.

803 11th Ave.

Sunnyvale, CA 94089

http://www.interwoven.com

Printed in the United States of America

Release 5.5.1

Part # 40-00-20-25-00-551-100

Table of Contents

About This Book

The *DataDeploy User's Guide* is intended for experienced DataDeploy™ users. This book describes how to access and use the DataDeploy portion of the Interwoven administration graphical user interface (GUI). The Interwoven® administration GUI is a Web application accessible from any system with a browser that is compatible with TeamSite (see the *TeamSite Administration Guide* for a matrix of browser compatibility).

Online Documentation Errata

Additions and corrections to this document are available in PDF format at the following site. Browse to the download and release notes directories.

`http://support.interwoven.com`

Notation Conventions

This manual uses the following notation conventions:

Convention	Definition and Usage
Bold	Text that appears in a GUI element (for example, a menu item, button, or element of a dialog box) and command names are shown in bold. For example: Click **Edit File** in the Button Bar.
Italic	Book titles appear in italics. Terms are italicized the first time they are introduced. Important information may be italicized for emphasis.

Convention	Definition and Usage
`Monospace`	Commands, command-line output, and file names are in monospace type. For example: The `iwextattr` command-line tool allows you to set and look up extended attributes on a file.
`Monospaced italic`	Monospaced italics are used for command-line variables. For example: `iwckrole` *`role user`* This means that you must replace *`role`* and *`user`* with your values.
`Monospaced bold`	Monospaced bold represents information you enter in response to system prompts. The character that appears before a line of user input represents the command prompt, and should not be typed. For example: **`iwextattr -s project=proj1 //IWSERVER/default/main/dev/WORKAREA/andre/products/index.html`**
`Monospaced bold italic`	Monospaced bold italic text is used to indicate a variable in user input. For example: **`% iwextattr -s project=`*`projectname workareavpath`* means that you must insert the values of *`projectname`* and *`workareavpath`* when you enter this command.
`[]`	Square brackets surrounding a command-line argument mean that the argument is optional.
`|`	Vertical bars separating command-line arguments mean that only one of the arguments can be used.

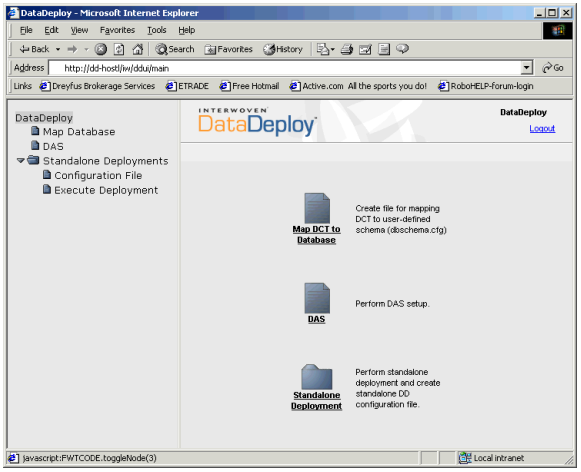
Chapter 1

Introduction

Interwoven administration server software components generate and manage the administration graphical user interfaces (GUIs) for Interwoven products. This book is limited to the DataDeploy portion of that GUI framework. The DataDeploy administration GUI is a Web application accessible from any system with a compatible browser.

The DataDeploy administration GUI provides an easy-to-use interface that enables you to:

- Map data capture templates (DCTs), and DTDs for data capture templates (DCT DTDs), to database schemas.
- Configure Data Auto-Synchronization (DAS).
- Perform standalone deployments.



Initial screen after login.

Requirements

Ensure that the following requirements are met. See the *DataDeploy Release Notes* for a list of TeamSite and TeamSite Templating versions that are compatible with this release of DataDeploy:

- TeamSite® must be installed.
- TeamSite Templating must be installed.
- End users must be listed in the TeamSite Master role file.

Browser Compatibility

The following table shows compatibility with Microsoft® Internet Explorer on various platforms (same as TeamSite 5.0.1). Netscape® will be supported in a later release of DataDeploy:

	Internet Explorer
Windows 95, 98, and NT	4.x-5.5
Windows 2000	5.0-5.5

Supported Databases

The DataDeploy administration GUI supports deployments to the following types of databases. It does not support deployments to wide tables, narrow tables, nor XML files:

- Oracle® 8i
- IBM DB2 UDB 7.1
- Microsoft SQL Server 7 and 2000

How to Access the GUI

You can log in to the DataDeploy administration GUI by pointing your browser to:

`http://dd-host/iw/ddui`

Use your TeamSite user name and password to log in.

Installation

The DataDeploy administration GUI is installed through the DataDeploy installer. When you install DataDeploy, select the option that installs the administration GUI.

A Note About Path Names

All paths to files and workareas entered through the DataDeploy administration GUI must be relative to TeamSite. For example:

```
/default/main/WORKAREA/wa1/templatedata/internet/medical/dbschema.cfg
```

is a valid path, while the following is not:

```
Y:/default/main/WORKAREA/wa1/templatedata/internet/medical/dbschema.cfg
```

Chapter 2

Mapping to a Database Schema

The DataDeploy administration GUI supports mapping data capture templates (DCTs) that are based on the Interwoven 4.5/5.0 DTD standards to a database schema. A DCT is an XML file called `datacapture.cfg`. The GUI does not support mapping DCTs that are not based on Interwoven DTDs.

The DataDeploy administration GUI also supports mapping DTDs for data capture templates (DCT DTDs) that are based on non-Interwoven DTDs.

Database schema files (`dbschema.cfg`) are files used by DataDeploy to specify how data is stored in a relational database.

The ability to map content based on any DCT or DCT DTD to a user-defined database schema provides great flexibility to users who require that data from TeamSite be deployed to normalized tables. By eliminating the need to edit XML and providing a tool for auto-generating `dbschema.cfg` files, the DataDeploy administration GUI greatly simplifies the process of mapping data capture files to database schemas.

To create user-defined schemas, elements in DCT or DCT DTD are mapped to columns in database tables. In the context of the DataDeploy administration GUI, these tables are referred to as "groups."

Process Flow

The following is an outline of end-user and system actions that take place when data capture files are mapped to database schemas:

- End users log in to the GUI and navigate to the Map Database section.

- End users set the mapping parameters by specifying the following:
  - **Source**: Specify which file is to be used as the source for the mapping and whether that file is a `datacapture.cfg` file (DCT based on Interwoven 4.5/5.0 standards) or a DCT DTD.
  - **Destination**: Specify whether they want to map to a new schema or map to an existing one.
  - **Database**: Specify the type of database.
  - **Map Type**: Specify whether to begin the process with a new, blank map or an auto-generated new map. Auto-generated maps provide a convenient starting point because groups are automatically created from the elements of the source file. These groups and their elements can be modified.
- Having set the mapping parameters, end users continue by creating groups and mapping elements in the source file to columns in those groups.

  Notes:

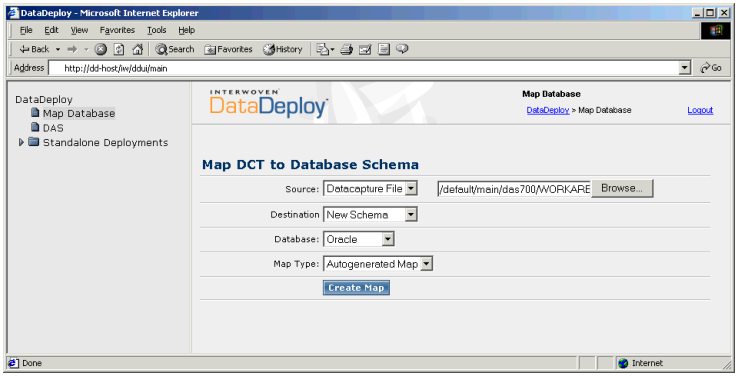
  - Each schema must contain exactly one root group. It is recommended that end users create and save the root group before creating other groups because the GUI populates drop-down menus with the group and column names of saved groups. It is helpful to end users to have that root group information available when they create subsequent groups.

    The root group is the parent to all other groups in the schema.
  - When creating columns, end users must (a) select an element from the source file, (b) select the column into which they want to copy the element, and (c) copy the element into that column. This select-and-copy method ensures that item names in schemas are identical to item names in the source file. Changing item names would prevent DataDeploy from deploying DCRs correctly. Additionally, the select-and-copy method ensures the transfer of other important "invisible" data from the source file to the schema.
  - The DataDeploy administration GUI does not validate end-user input. If end users enter invalid input, DataDeploy will display errors at deployment time.
- When a group and corresponding columns have been specified, end users save the group. It is then displayed in the **Mapped Tables** pane of the GUI.
- When end users finish creating or editing groups, they view the resulting `dbschema.cfg` file and save it to the workarea location they want.

Setting Mapping Parameters

To set the general parameters for mapping a `datacapture.cfg` file or a custom DTD:

1. Log in to the DataDeploy administration GUI. See “How to Access the GUI” on page 9.
2. Navigate to the Map Database section:



3. Specify whether the file you want to use is a `datacapture.cfg` file or a DTD by making a selection from the **Source** drop-down menu.

   Notes: Data capture files must be based on the Interwoven 4.5/5.0 DTD.

   Selecting **Custom DTD** implies that a non-Interwoven DTD will be specified.

4. Click **Browse**.

   The Select File dialog box is displayed. You can browse only the directories and files that are within your workarea.

5. Browse to the location of the file that you want to map. Double click to browse directories; single click to select files.
6. Use the **Destination** drop-down menu to specify whether you want to map to a new schema or map to an existing one.

7. From the **Database** drop-down menu, select the type of database where your schema will be mapped.
8. Use the **Map Type** drop-down menu to specify whether you want to create a new, blank map or use an auto-generated new map as a starting point.
9. Click **Create Map**.

What happens after you click **Create Map** depends on the selections you made when you set the mapping parameters:

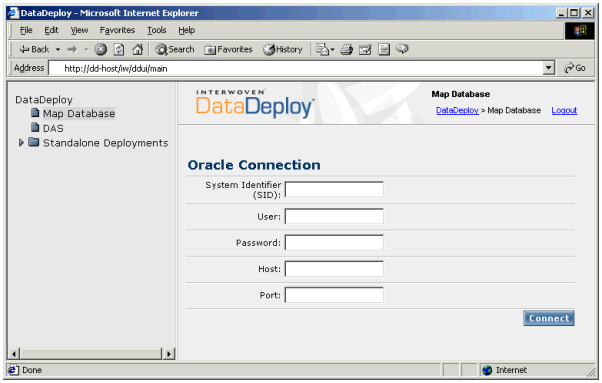
- If you chose to create a new schema, the Map Source File to a New Database Schema screen is displayed. See "Creating a New Database Schema" on page 16.
- If you chose to edit an existing schema, the database connection screen is displayed. See "Mapping to an Existing Database Schema" on page 14.

Mapping to an Existing Database Schema

To map to an existing schema:

1. Set the mapping parameters according to your needs (see "Setting Mapping Parameters" on page 13). Ensure that the **Destination** parameter is set to Existing Schema.
2. Click **Create Map**.

   The database connection screen is displayed:



Connection screen as displayed when Oracle is chosen as the database type

3. Enter the System Identifier (Oracle), or the database name (Microsoft SQL server, DB2).
4. Enter your user name and password.
5. Enter the name of the system where the database is located.
6. Enter the port number where the database is located.
7. Click **Connect**.

   A status dialog box is displayed as the system connects to the database. A temporary connection is made to retrieve the database tables. After the tables have been retrieved, the mapping tool is displayed and the connection is terminated.
8. In the mapping tool, select the table you want to map.

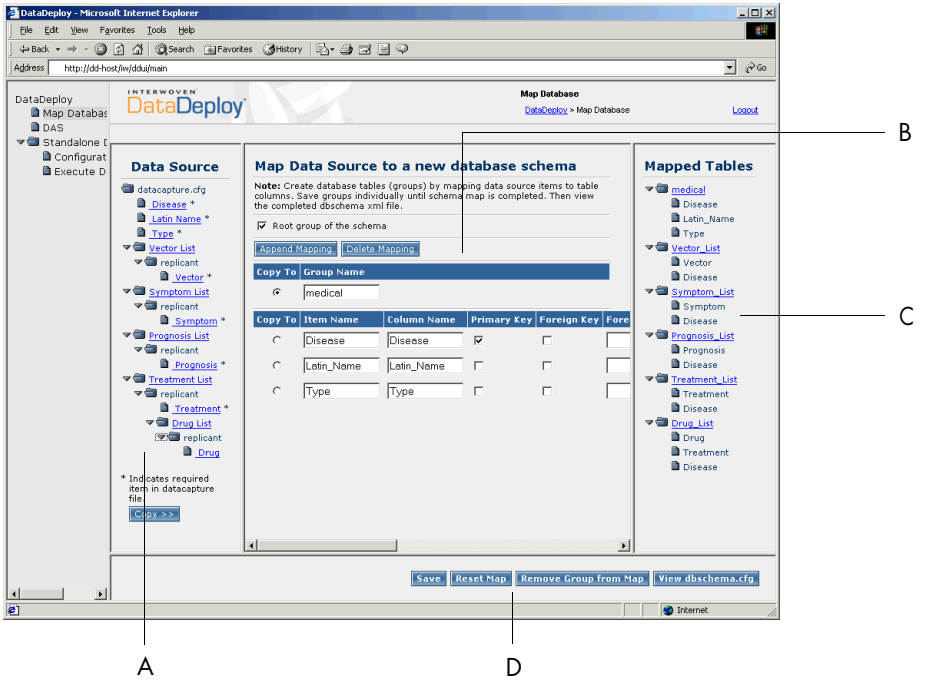
   Fields in the mapping tool are populated with the properties of that table. See the description of the mapping tool (section B of the GUI) in “Creating a New Database Schema” on page 16.

Creating a New Database Schema

To create a new schema:

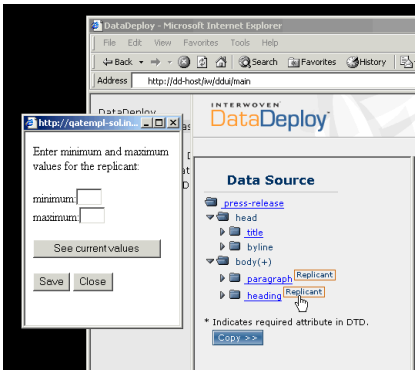
1. Set the mapping parameters according to your needs (see “Setting Mapping Parameters” on page 13). Ensure that the **Destination** parameter is set to New Schema.
2. Click **Create Map**.

   The mapping screen is displayed. The screen has four panels which are described below:



A. The Data Source panel displays the elements of the XML source file in a tree format. XML nodes contain nested elements, and are indicated by folder icons. Click the arrow next to a node to display its nested elements.

If you have chosen a custom DTD as the source file, **Replicant** buttons are displayed next to replicant elements in the source tree. Click these buttons to open a dialog box that enables you to view the existing values that specify the minimum and maximum instances of the replicant, and to set different minimum and maximum values. (When you save different values, the new values are transferred to the `dbschema.cfg` file. After you save the group, click **View dbschema.cfg** if you want to verify that your changes are reflected there:)

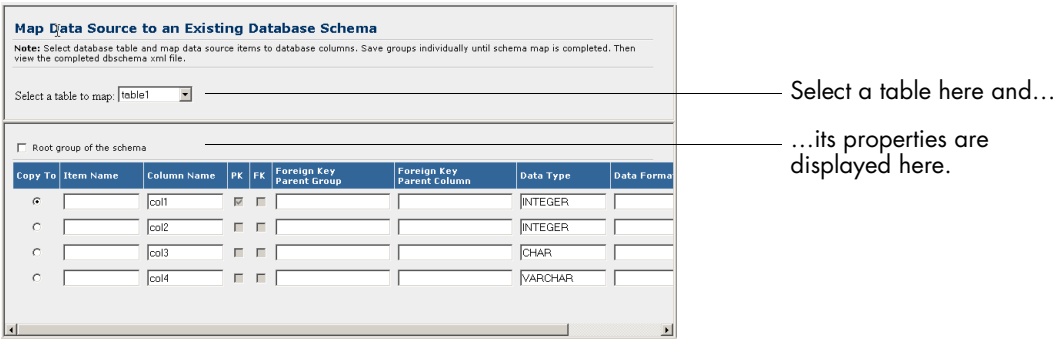


When mapping a custom DTD, you can specify the minimum and maximum number of instances for replicants.

Select elements in A and copy them to input fields in B to map those elements to columns in a group. Some elements are not mappable (Replicant, for example); those elements cannot be selected.

B. The Map Data Source to a New Database Schema panel contains the buttons and input fields that enable you to specify how elements from the source file map to columns in the groups you create.

   If you chose to work with existing tables, the GUI enables you to select the table you want. When you select a table, its properties are displayed in the fields of the mapping tool:



Elements can be copied from A to B. Also, saved mappings listed in C can be edited when they are populated into B (see C below).

C. When you save groups that you create in B, they are displayed in tree format in the Mapped Tables panel. Groups listed in C are hyper linked so that when you click on a group, B is populated with the mappings in that group.

Note: If you save (or remove) a group and it is not immediately displayed in the Mapped Tables panel, right click in that panel and choose **Refresh** from the Internet Explorer context menu. When that panel is refreshed the group is displayed.

D. This panel contains buttons that enable you to:

- **Save**: Saves your mappings.
- **Reset Map**: Clears the input fields.
- **Remove Group from Map**: Removes selected groups from the schema.
- **View dbschema.cfg**: Displays the actual XML file, dbschema.cfg, that is the result of your mappings.

3. Create a root group. The root group is the parent to all other tables in the schema. Each schema must contain exactly one root table. It is recommended that you create and save a root group first because that enables the GUI to populate the drop-down menus with that table's group and column names.

   To specify a group as the root table, check the **Root group of the schema** check box.

4. Create additional groups according to your needs:

   a. Enter a name for the group in the **Group Name** field, or copy an element from the data source to that field.

   b. Create columns by copying elements (see "Process Flow" on page 11 for details on the select-and-copy method.) to the **Item Name** and **Column Name** fields and specifying the following. See your *DataDeploy Administration Guide* for details about these items:

      – **Column Name**: Specifies the names of the column.
      – **Primary Key**: Specifies whether the data stored in that column is a primary key.
      – **Foreign Key**: Specifies whether the data stored in that column is a foreign key.
      – **Foreign Key Parent Group**: Specifies the parent group of the data if it is a foreign key. This drop-down menu contains the group names of groups in the schema that have been saved.
      – **Foreign Key Parent Column**: Specifies the parent column of the data if it is a foreign key. This drop-down menu contains the column names of groups that have been saved.
      – **Data Type/Length**: Specifies the data type and maximum length of data in that column.
      – **Data Format**: Specifies the format in which data is stored in that column (for example, `dd-mm-yyyy`).
      – **Not Null**: Specifies whether the column data can be null.
      – **Is-URL**: Specifies whether the column data is a URL. If the column data is specified as a URL, DataDeploy deploys the content where that URL points, not the URL itself.

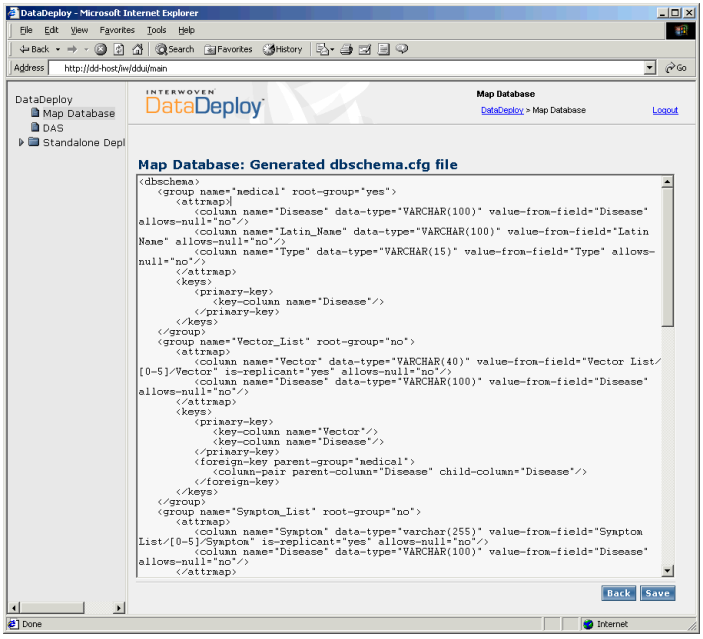
   c. Use the **Append Mapping** or **Delete Mapping** buttons to add or delete columns from a group.

5. When you complete a group, click **Save**.

   The group is displayed in the Mapped Tables pane. If it is not immediately displayed there, right click in that panel and choose **Refresh** from the Internet Explorer context menu. When the Mapped Tables panel is refreshed the group is displayed.

6. When you have finished creating and saving groups, click **View dbschema.cfg**.

   The XML file, `dbschema.cfg`, is displayed. See your *DataDeploy Administration Guide* for details about `dbschema.cfg` files.



7. Save the `dbschema.cfg` file:

   a. Click **Save**. Scroll to the bottom of the main window (as opposed to the window that displays the file) if you do not see the **Save** button.

      The Select File dialog is displayed.

   b. Browse to the location where you want to save the file. Double click to browse directories; single click to select files.

   c. Click **OK**.

Chapter 3

Configuring Database Auto-Synchronization

From the DAS section of the DataDeploy administration GUI, you can:

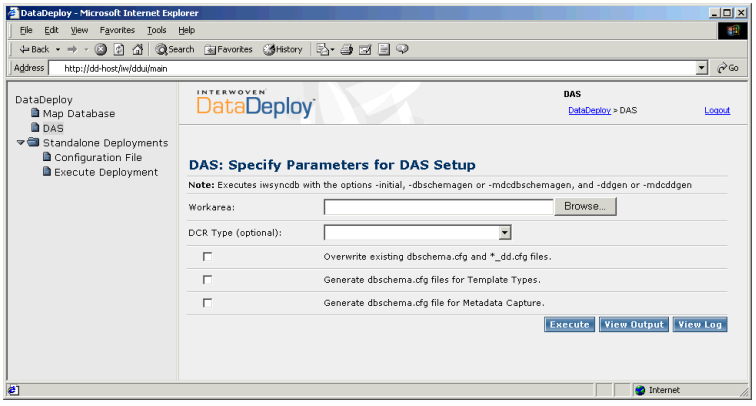
- Specify parameters for Database Auto-Synchronization (DAS).
- Start the DAS module.
- View the standard output of DAS deployments.
- View deployment log files.

Consult your *DataDeploy Administration Guide* for complete details about the DAS feature and its components.

Specifying DAS parameters

To specify DAS parameters:

1. Log in to the DataDeploy administration GUI. See "How to Access the GUI" on page 9.
2. Navigate to the DAS section:



3. Click **Browse**.

   The Select File dialog box is displayed. You can browse only the directories and files that are within your workarea.

4. Browse to the workarea that you want. Double click to browse directories; single click to select them.

5. (Optional) Select the type of DCR you want.

6. Check the check boxes next to the options that you want.

7. Click **Execute**.

   The DAS module is started with the options you specified, and a status dialog box is displayed.

Viewing Output

To view the standard output of DAS deployments:

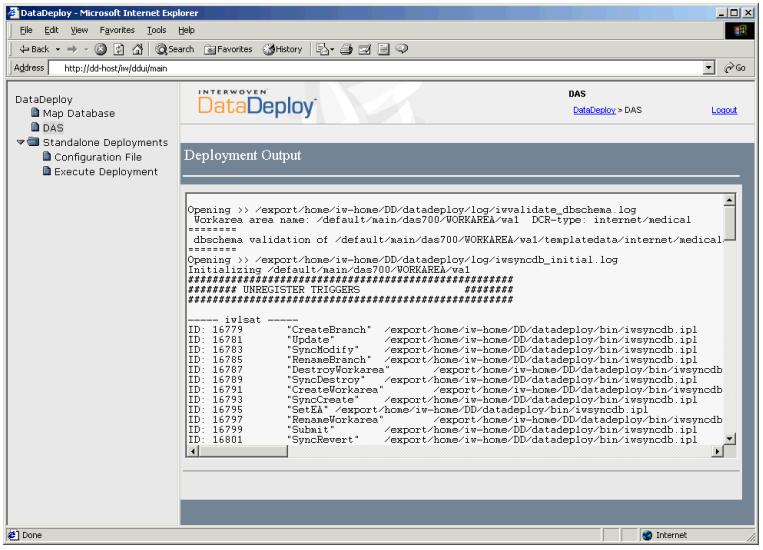
1. Execute the `iwsynchdb.ipl` script. See “Specifying DAS parameters” on page 21.

   A status dialog box is displayed.

2. Click **OK** to close the status dialog box.

3. Click **View Output** in the Specify Parameters for DAS Setup screen.

   The Deployment Output window is displayed:

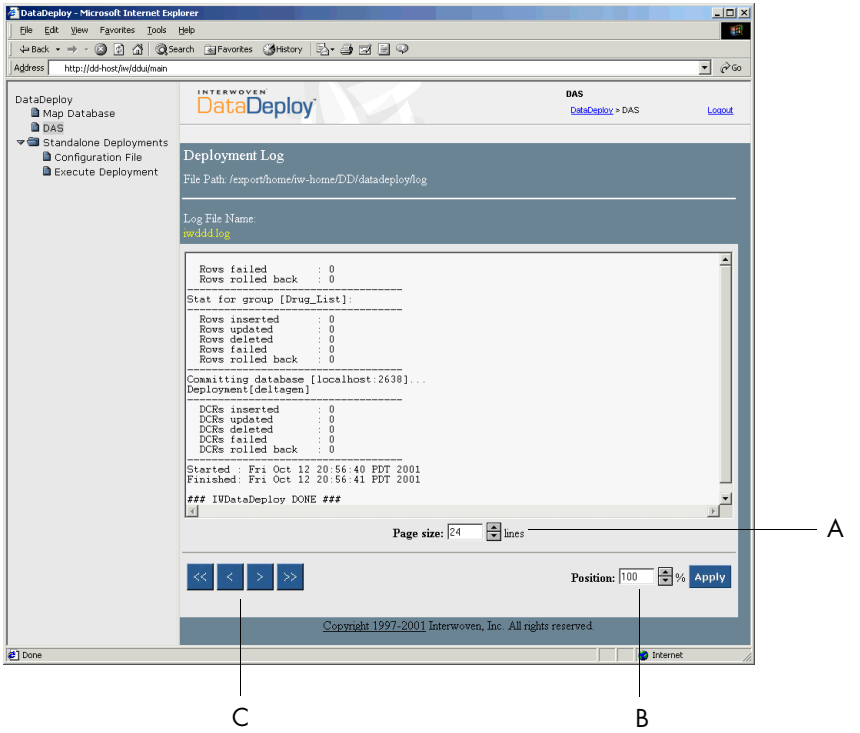


The output from your system might be different from the sample output displayed here.

Viewing Deployment Log Files

1. Log in to the DataDeploy Administration GUI. See “How to Access the GUI” on page 9.
2. Navigate to the DAS section.
3. Click **View Log**.

   The Deployment Log window is displayed. Controls for navigating the file are described below:



A. Sets the maximum number of lines to be displayed in the window.

B. Positions you in a general area of the log file. For example: If you are searching for entries in a large log file and you know that the entries exist in the earliest 1/3 of the file, you can begin your search by entering **33%** in the **Position** field. When you click **Apply**, the most recent page (in this case, a set of 24 lines) of that earliest 1/3 portion of the log file. Use the buttons described in C to pinpoint the entries you want.

C. <<: Displays the earliest page of entries.

<: Displays the previous page of entries.

>: Displays the next page of entries.

>>: Displays the most recent page of entries.

Chapter 4

Performing Standalone Deployments

From the standalone deployment section, you can:

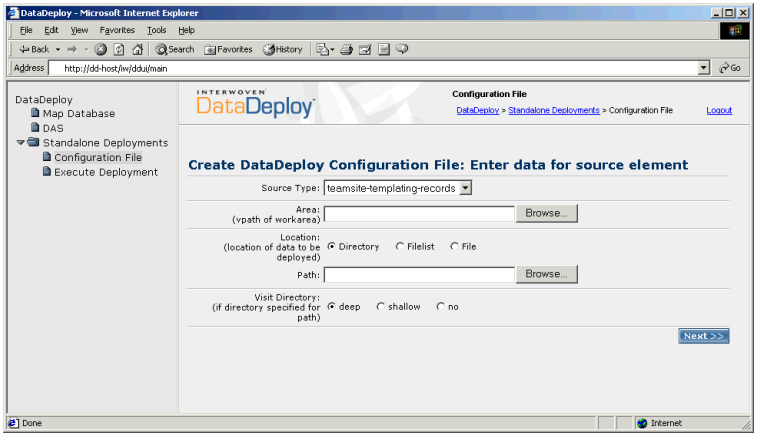
- Create a DataDeploy configuration file.
- Deploy data content records (DCRs) and TeamSite metadata into a database.

This chapter does not describe the items listed above. Consult your *DataDeploy Administration Guide* for complete details about DataDeploy configuration files and standalone deployments.

Creating a DataDeploy Configuration File

To create a DataDeploy configuration file:

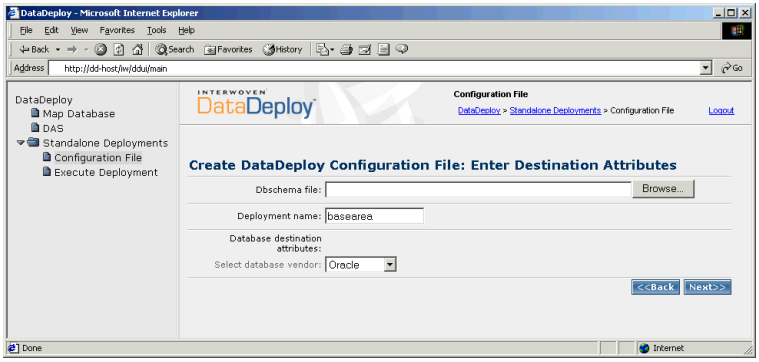
1. Log in to the DataDeploy administration GUI. See "How to Access the GUI" on page 9.
2. Navigate to the Standalone Deployments > Configuration File section:



3. Specify the following items:
   - **Source Type**: Specifies whether you want to deploy DCRs (teamsite-templating-records) or TeamSite metadata (teamsite-extended-attributes).
   - **Area**: Specify the vpath to the workarea from which you want to deploy data.
   - **Location**: Specify the location of the data you want to deploy and whether the data is located in a directory, filelist, or file. If select **Directory**, the last set of options in the screen enable you to specify how deep into a directory structure you want DataDeploy to search when it searches for deployment data. See **Visit Directory**, below.
   - **Visit Directory**: Specify whether you want DataDeploy to search recursively through the directory, and all its subdirectories, that you chose (**deep**), or search only the top level of the chosen directory (**shallow**). Select **no** if you do not want DataDeploy to search for deployment data in the location you specified.

4. Click **Next**.

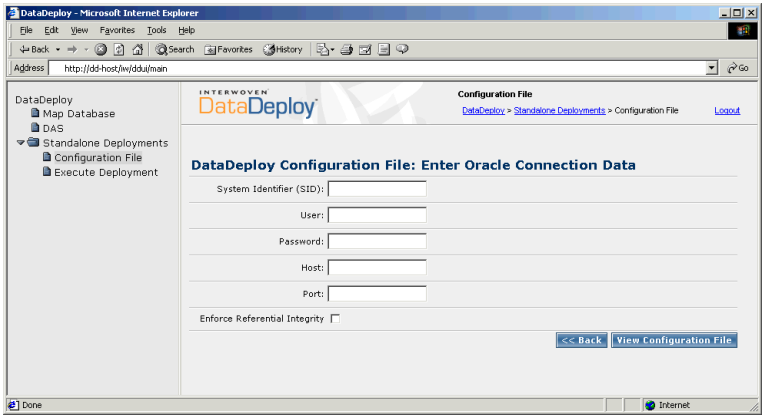
   The Enter Destination Attributes screen is displayed:



5. Specify the following items:
   - **Dbschema file**: Specify the `dbschema.cfg` file you want to use. If you saved a `dbschema.cfg` file during the current browser session through the Map Database section of the GUI, the location of that file is displayed in the field.
   - **Deployment name**: Enter a name for the deployment (basearea is the default name).
   - **Database destination attributes** (database vendor): Enter the type of database.

6. Click **Next**.

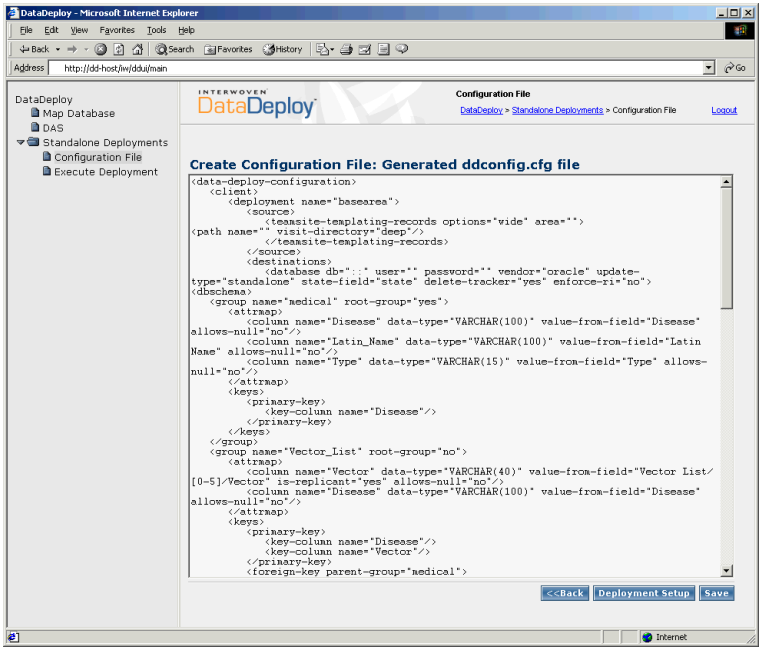
The database connection screen is displayed:



Connection screen as displayed when Oracle is chosen as the database type

7. Enter the System Identifier (Oracle), or the database name (Microsoft SQL server, DB2).
8. Enter your user name and password.
9. Enter the name of the system where the database is located.
10. Enter the port number where the database is located.
11. (Optional) Check the **Enforce Referential Integrity** check box if you want the constraints set by the primary and foreign keys enforced during deployment and while creating tables.
12. Click **View Configuration File**.

    The DataDeploy configuration file is displayed. The following is a sample file; the one generated from your input might look different:



Note: Modifications made to the file in this window can not be saved. Modify the file through the previous steps, or through the TeamSite WebDesk interface.

13. Click **Save**.

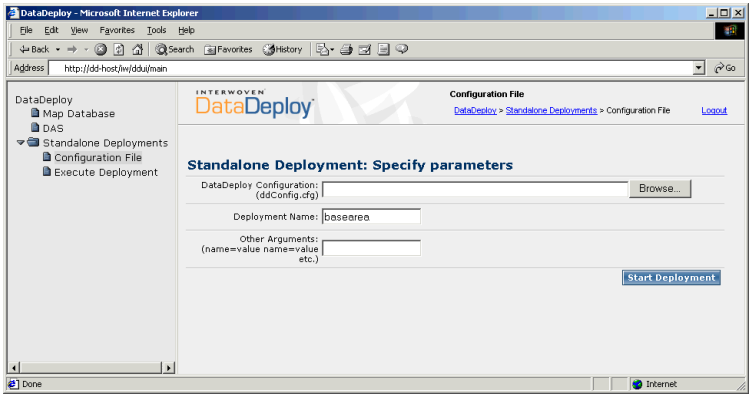
    The Select File dialog box is displayed.

14. Browse to the location where you want to save the file and click **OK**.

After you have saved the DataDeploy configuration file you can click **Deployment Setup** to continue by executing a deployment (see “Executing Deployments” on page 31.

Executing Deployments

You can execute a deployment from the Standalone Deployments section of the GUI:



The Standalone Deployments section of the GUI.

You can get to this screen by:

- Navigating to Standalone Deployments > Execute Deployments in the navigation pane.
- Completing the steps in "Creating a DataDeploy Configuration File" on page 27 and clicking on **Deployment Setup** in the Generated ddconfig.cfg File screen.

To execute a deployment:

1. In the Specify Parameters screen, specify the following items:
   - **DataDeploy Configuration**: Specify the location of the DataDeploy configuration file that you want to use. If you saved a DataDeploy configuration file during the current browser session through the Configuration File section of the GUI, the location of that file is displayed in the field.
   - **Deployment Name**: Specify a name for the deployment. If you specified a name in the Configuration File section of the GUI, that name is displayed in the field.
   - **Other Arguments**: Enter additional arguments.The arguments must be entered in `name=value` format. These arguments are appended to the `iwdd.ipl` command that is run when the deployment is started.

2. Click **Start Deployment**.

   When the deployment is complete the output is displayed. The following is sample output; the one generated from your input might look different: